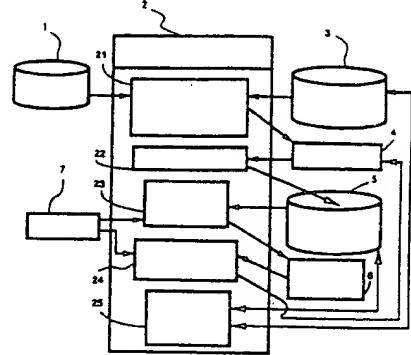


(54) KNOWLEDGE PROCESSING SY  
 (11) 3-41519 (A) (43) 22.2.1991 JP  
 (21) Appl. No. 64-175467 (22) 10.7.1989  
 (71) HITACHI LTD (72) NAOKI KATO(1)  
 (51) Int. Cl<sup>s</sup>. G06F9/44

**PURPOSE:** To attain highly efficient inference and to prevent an erroneous conclusion from being disabled to arrive at a required goal by providing a system with a means for temporarily executing and evaluating a rule to select and execute an optimum rule.

**CONSTITUTION:** A 2nd working memory 5, an evaluated result storing area 6 and an evaluation knowledge part 7 storing an evaluation function, a threshold, etc., are added to a conventional constitution. An executed result evaluating means 23 for evaluating an executed result in the 2nd working memory 5 based upon the contents of the evaluation knowledge part 7 and storing the evaluated result in an evaluated result storing area 6 and a data restoring/updating means 25 are added to the inference part 2. In the case of selecting one instantiation from a conflict set 4, a rule is executed by a rule executing means 22, the executed result is evaluated by the evaluation knowledge part 7 and the evaluating means 23 and an instantiation with high evaluation is selected by an instantiation selecting means 24. Consequently, the efficiency of inference for obtaining a required goal can be improved and role formation and rule base construction can easily be executed.

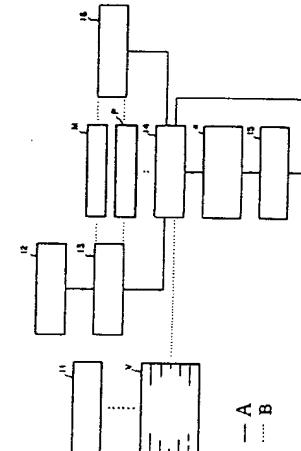


1: rule base. 3: 1st working memory. 4: conflict set. 21: condition collating and conflict set forming means. 22: rule executing means

(54) INTERRUPTION CONTROL SYSTEM  
 (11) 3-41520 (A) (43) 22.2.1991 (19) JP  
 (21) Appl. No. 64-175572 (22) 10.7.1989  
 (71) NEC CORP (72) HISAYUKI AOKI  
 (51) Int. Cl<sup>s</sup>. G06F9/46

**PURPOSE:** To prevent interruption processing having no relation to its resources from being unnecessarily delayed without masking it by independently masking each interruption processing.

**CONSTITUTION:** The interruption control system is provided with an interruption vector setting means 11, an interruption cause inputting means 12 and an interruption mask deciding means 13 for checking the masking state of an interruption generated from an interruption mask register M and deciding whether the interruption processing is to be held or not. An interruption handler starting means 14 also is provided, the interruption mark register is restored by an interruption handler ending means 15 and whether or not interruption processing held by its value and the value of the interruption holding register is used for the interruption processing is decided and the interruption processing is ended or branched to the means 14 again. Consequently, the delay of the interruption processing can be suppressed to the required minimum.

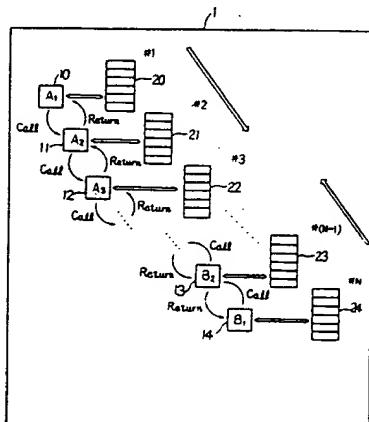


16: interruption mask changing means. A: processing flow. B: data reference, updating. P: interruption holding means. V: interruption vector. a: each interruption handler

(54) REGISTER WINDOW SYSTEM  
 (11) 3-41521 (A) (43) 22.2.1991 (19) JP  
 (21) Appl. No. 64-175571 (22) 10.7.1989  
 (71) NEC CORP (72) HIROBUMI KOMIYAMA  
 (51) Int. Cl<sup>s</sup>. G06F9/46

**PURPOSE:** To shorten process switching time by allocating two process groups to plural register sets so that one of them is a forward direction and the other is the reverse direction.

**CONSTITUTION:** The register set groups #1 to #N are used from two directions, i.e. the direction of #1 → #2 → #3 → and the direction of #N → #N-1 →. Thereby, two different process groups, i.e. process A<sub>1</sub> → process A<sub>2</sub> → process A<sub>3</sub> → and process B<sub>1</sub> → process B<sub>2</sub> → can be simultaneously processed and the two process groups can be continuously processed without stopping the execution of one process group or without executing the retreat/restoration, i.e. save/restore of the register sets at the time of switching the process group. Thus, the process switching time can be shortened.



1: arithmetic processor

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 03-041520  
(43)Date of publication of application : 22.02.1991

---

(51)Int.CI. G06F 9/46

(21)Application number : 01-175572 (71)Applicant : NEC CORP  
(22)Date of filing : 10.07.1989 (72)Inventor : AOKI HISAYUKI

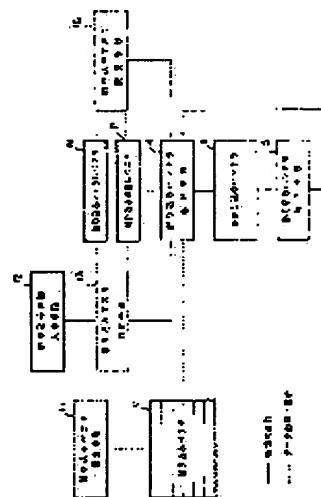
---

## (54) INTERRUPTION CONTROL SYSTEM

### (57)Abstract:

**PURPOSE:** To prevent interruption processing having no relation to its resources from being unnecessarily delayed without masking it by independently masking each interruption processing.

**CONSTITUTION:** The interruption control system is provided with an interruption vector setting means 11, an interruption cause inputting means 12 and an interruption mask deciding means 13 for checking the masking state of an interruption generated from an interruption mask register M and deciding whether the interruption processing is to be held or not. An interruption handler starting means 14 also is provided, the interruption mark register is restored by an interruption handler ending means 15 and whether or not interruption processing held by its value and the value of the interruption holding register is used for the interruption processing is decided and the interruption processing is ended or branched to the means 14 again. Consequently, the delay of the interruption processing can be suppressed to the required minimum.



---

## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑨ 日本国特許庁 (JP) ⑩ 特許出願公開  
⑪ 公開特許公報 (A) 平3-41520

⑫ Int. Cl.<sup>6</sup>  
G 06 F 9/46

識別記号 庁内整理番号  
311 E 8945-5B

⑬ 公開 平成3年(1991)2月22日

審査請求 未請求 請求項の数 1 (全8頁)

⑭ 発明の名称 割り込み制御方式

⑮ 特 願 平1-175572  
⑯ 出 願 平1(1989)7月10日

⑰ 発明者 青木 久幸 東京都港区芝5丁目33番1号 日本電気株式会社内  
⑱ 出願人 日本電気株式会社 東京都港区芝5丁目7番1号  
⑲ 代理人 弁理士 芦田 坦 外2名

明細書

1. 発明の名称

割り込み制御方式

2. 特許請求の範囲

1. 各原因に対応する割り込みの処理を行う割り込みハンドラと、割り込みの原因に対応して、割り込みを許可するか否かを示す割り込みマスクレジスタと、割り込みの原因に対応して、割り込みを保留するための割り込み保留レジスタと、前記割り込みハンドラの先頭アドレスと該割り込みハンドラの初期割り込みマスクレジスタの値とを記憶する割り込みベクタとを用いて、割り込みを制御する方式に於いて。

割り込みが発生した時に、その原因入手する割り込み原因入手手段と、

前記割り込みマスクレジスタから発生した割り込みのマスク状態を調べ、割り込みを保留するか否かを判定し、前記発生した割り込みがマスクさ

れているときには、前記割り込み保留レジスタに当該発生した割り込みを保留し、前記発生した割り込みがマスクされていないときには、割り込み許可信号を出力する割り込みマスク判定手段と、

前記割り込み許可信号に応答して、前記割り込みマスクレジスタの内容を退避し、前記割り込みベクタを参照して前記発生した割り込みの処理を行う割り込みハンドラを呼び出す割り込みハンドラ起動手段と、

前記呼び出された割り込みハンドラの処理が終了したとき、前記割り込みマスクレジスタを復元し、該復元された値と前記割り込み保留レジスタの値とから保留されている割り込み処理で処理可能なものがあるか否かを判定し、割り込み処理を終了するかまたは再度前記割り込みハンドラ起動手段に分岐する割り込みハンドラ終了手段と、

外部からの変更要求に応答して、前記割り込みマスクレジスタの状態を変更する割り込みマスク変更手段と、

を有することを特徴とする割り込み制御方式。

## 3. 発明の詳細な説明

## 【産業上の利用分野】

本発明は、計算機システムにおいて、割り込み処理ルーチンと通常のルーチンとの間の資源排他制御方式に関する。

## 【従来の技術】

ある資源を操作中に割り込みが発生して、その割り込みに対する処理ルーチンでその資源を操作すると、両者の操作内容によっては矛盾が発生する。例えば、あるチェイン上の構造体を別のチェインに移動させるような処理は、元のチェインから外した時点で割り込まれ、その割り込み処理ルーチン内でも同じ構造体に対するチェイン操作を行えば、チェインに乱れが生じる。このような現象を防ぐためには、通常のルーチンと割り込み処理ルーチンとの間で資源排他制御を行う必要がある。

一般に、資源排他制御の手段としては、ロック方式やセマフォによる方式などがある。これらの手法は、すでに使われている資源を別のプログラ

ムが操作しようとした時に、そのプログラムの実行を一時中断し、先にその資源を使っていたプログラムの実行完了を待ち合わせることにより、同一の資源に対する同時アクセスを排除するものである。しかし、これらの手法を通常ルーチンと割り込み処理ルーチンとの資源排他制御に使うと、割り込み処理ルーチンの中で実行が停止してしまう。通常ルーチンは割り込み処理から戻らないと動作できないので、永久にプロセッサが停止してしまうか、あるいは、無限ループになってしまいます。つまり、これらの手法はプロセス間(タスク間)の資源排他制御方式であって、通常ルーチンと割り込みルーチンのような場合には適用できない。

通常ルーチンと割り込みルーチンとの間の資源排他制御の手法としては、割り込みマスクによる方式がある。割り込みがマスクされていると、割り込みが発生しても割り込みマスクが解除されるまで、割り込み処理を保留するという方式である。通常ルーチンにおいて、割り込み処理ルーチンで操作される資源を使う時は、この割り込みマスク

をセットすればよい。割り込みマスクがセットされている期間は、割り込み処理ルーチンが動かないのに、問題の資源を自由に操作できる。

この割り込みマスク方式の発展した方式として割り込みレベル制御方式がある。この方式は、割り込みマスク(1ビット)の代わりに割り込みレベル(数ビット)なる領域を使う。原因の違う各割り込みは、それぞれ対応する割り込みレベルを持っている。割り込みレベルをある値に設定すると、そのレベル以下の割り込みが発生しても、割り込みレベルが低くなるまで割り込み処理は保留される。従って、割り込みレベルの値を変更することによって、資源の排他制御が可能となる。割り込み処理を保留する事によって資源の排他制御を行う点は、割り込みマスク方式と同じである。しかし、この方式では、ある割り込みレベルを設定しても、そのレベルより高い割り込みレベルに対応する割り込みの割り込み処理は保留されない。つまり、割り込みマスク方式と違って、すべての割り込み処理が保留されるわけではない。

## 【発明が解決しようとする課題】

割り込みマスク方式では割り込みがマスクされていると、一切の割り込み処理ルーチンが動かなくなるという問題点がある。一般に、割り込みの原因はひとつのみではなく複数ある。従って、実際に発生した割り込みの原因によっては、問題の資源を使わない場合がある。このような割り込みはマスクする必要がない。割り込みマスク方式では、このような割り込み処理も無条件に保留にしてしまう。延滞させる必要のない割り込み処理までも延滞させてしまう。

同様に、割り込みレベル制御方式でも、設定されているレベルより低いレベルに対応する割り込みの処理はすべて延滞してしまう。一般には、あるレベルに対応する割り込み処理で使う資源を、そのレベルより低いレベルに対応する割り込み処理で必ず使うわけではない。この場合も、延滞させる必要のない割り込み処理までも延滞させている事になる。

一般に、割り込み処理は割り込みが発生した直

後に行うのが型ましい。割り込み発生から時間が経過すれば、それだけシステム内外の状況が変化してしまうし、システムの応答性能も劣化する。たとえば、一定間隔でタイマ割り込みを発生させるような場合は、割り込み処理を保留しているうちに次のタイマ割り込み時刻になってしまふ。このような観点から見れば、割り込みマスク方式も割り込みレベル制御方式も、外部事象に対する応答性能等をある程度犠牲にして、資源の排他制御を行っていると言える。さらに、犠牲という点から見れば、常にすべての割り込み処理が保留されてしまうわけではない分だけ、割り込みレベル制御方式の方が、割り込みマスク方式よりも進歩していると考えられる。

本発明の目的は、通常ルーチンと割り込み処理ルーチンとの資源排他制御において、割り込み処理の経済を必要最小限で済ませ、応答性能等の犠牲をなるべく少なくする事ができる割り込み制御方式を提供することにある。

#### [課題を解決するための手段]

次に本発明について図面を参照して詳細に説明する。

第1図を参照すると、本発明の一実施例による割り込み制御方式が適用される計算機システムは、割り込みベクタVに各原因に対応する割り込みの処理を行う割り込みハンドラの先頭アドレスと、その割り込みハンドラの初期割り込みマスクレジスタの値を設定する割り込みベクタ設定手段11と、割り込みが発生した時にその原因入手する割り込み原因入手手段12と、割り込みマスクレジスタMから発生した割り込みのマスク状態を調べ、割り込み処理を保留するか否かを判定する割り込みマスク判定手段13と、割り込みマスクレジスタMを待避し、割り込みベクタVを参照して、発生した割り込みの処理を行う割り込みハンドラaを呼び出す割り込みハンドラ起動手段14と、割り込みマスクレジスタMを復元し、その値と割り込み保留レジスタPの値とから保留されている割り込み処理で処理可能なものがあるか否かを判定し、割り込み処理を終了するかまたは再度割り

り込みによる割り込み制御方式は、割り込みベクタに各原因に対応する割り込みの処理を行う割り込みハンドラの先頭アドレスとその割り込みハンドラの初期割り込みマスクレジスタの値を設定する割り込みベクタ設定手段と、割り込みが発生した時にその原因入手する割り込み原因入手手段と、割り込みマスクレジスタから発生した割り込みのマスク状態を調べ割り込み処理を保留するか否かを判定する割り込みマスク判定手段と、割り込みマスクレジスタを待避し割り込みベクタを参照して発生した割り込みの処理を行う割り込みハンドラを呼び出す割り込みハンドラ起動手段と、割り込みマスクレジスタを復元しその値と割り込み保留レジスタの値とから保留されている割り込み処理で処理可能なものがあるか否かを判定し割り込み処理を終了するかまたは再度割り込みハンドラ起動手段に分岐する割り込みハンドラ終了手段と、割り込みマスクレジスタの状態を変更する割り込みマスク変更手段とを有している。

#### [実施例]

込みハンドラ起動手段14に分岐する割り込みハンドラ終了手段15と、割り込みマスクレジスタMの状態を変更する割り込みマスク変更手段16とから構成される。

次に本発明の動作について図面を参照して説明する。

第2図を参照すると、割り込みマスク判定手段13は、割り込みマスク判定ステップ21で、割り込みマスクレジスタ（第1図のM）の発生した割り込みに対応するビットを調べる。マスクされていない場合は、割り込みハンドラ起動手段14へ分岐する。マスクされている場合は、割り込み保留処理ステップ22で、割り込み保留レジスタ（第1図のP）の発生した割り込みに対応するビットをセットし、割り込み処理から戻る。

第3図を参照すると、割り込みハンドラ起動手段14は、割り込みマスク待避ステップ31で、その時の割り込みマスクレジスタ（第1図のM）の値をシステムスタックなどに待避し、割り込みマスク変更ステップ32で、割り込みベクタ（第

1図のV)から発生した割り込みに対応する割り込みハンドラの初期割り込みマスクレジスタ値を読み、この読まれた初期割り込みマスクレジスタ値と割り込みマスクレジスタ(第1図のM)の現在値との論理和を取り、その結果を割り込みマスクレジスタ(第1図のM)に設定し、割り込みハンドラ先頭アドレス入手ステップ33で、割り込みベクタから発生した割り込みに対応する割り込みハンドラの先頭アドレスを入手し、その割り込みハンドラに分岐する。

第4図を参照すると、割り込みハンドラ終了手段15は、割り込みマスク復元ステップ41で、前記割り込みハンドラ起動手段14でシステムマスクなどに待避された値を割り込みマスクレジスタ(第1図のM)に復元し、割り込み保留判定ステップ42で、割り込みマスクレジスタ(第1図のM)と割り込み保留レジスタ(第1図のP)とからマスクが解除されて処理が保留されている割り込みがあるか否かを検査する。マスクが解除され処理が保留されている割り込みがある場合は、

否かを検査する。マスクが解除され処理が保留されている割り込みがある場合は、割り込み原因選択ステップ53で、次に処理する割り込みを選び、割り込み保留解除ステップ54で、割り込み保留レジスタ(第1図のP)の割り込み原因選択ステップ53で選ばれた割り込みに対応するビットをクリアし、割り込みハンドラ起動手段14へ分岐する。マスクが解除され処理が保留されている割り込みがない場合は、割り込みマスク変更手段16を呼び出した所に戻る。

次に、第6図、第7図、及び第8図を参照して、本発明の割り込み制御方式の動作について説明する。

各割り込みハンドラが使用する資源を第7図の通りとする。第7図より、割り込みハンドラ $\alpha$ は割り込みハンドラ $\beta$ と割り込みハンドラ $\gamma$ と同じ資源を使用する。従って、割り込みハンドラ $\alpha$ が動作している期間は割り込みハンドラ $\beta$ と割り込みハンドラ $\gamma$ の起動を抑止しなければならない。他の割り込みハンドラについても同様に考えると、

割り込み原因選択ステップ43で、次に処理する割り込みを選び、割り込み保留解除ステップ44で、割り込み保留レジスタ(第1図のP)の割り込み原因選択ステップ43で選ばれた割り込みに対応するビットをクリアし、割り込みハンドラ起動手段14へ分岐する。マスクが解除され処理が保留されている割り込みがない場合は、呼び出し元判定ステップ45で、割り込み処理と割り込みマスク変更手段16とのいずれから呼び出されたかを判定し、割り込み処理から呼び出された場合は割り込み処理から戻り、割り込みマスク変更手段16から呼び出された場合は、割り込みマスク変更手段16を呼び出した所に戻る。

第5図を参照すると、割り込みマスク変更手段16は、割り込みマスク変更ステップ51で、指定された値を割り込みマスクレジスタ(第1図のM)に設定し、割り込み保留判定ステップ52で、割り込みマスクレジスタ(第1図のM)と割り込み保留レジスタ(第1図のP)とからマスクが解除されて処理が保留されている割り込みがあるか

第6図のような、割り込みベクタが必要であることがわかる。

さて、第8図に実際の動作例を示す。

時刻aで資源1を使用するために、割り込みマスク変更手段16を使って、割り込みAと割り込みBをマスクする。

しばらくして、時刻bで割り込みBが発生すると、割り込み原因入手手段12・割り込みマスク判定手段13が動作するが、割り込みBがマスクされているので、割り込みハンドラBは保留状態になり、呼び元の処理に戻る。

時刻cで割り込みCが発生すると、割り込みCはマスクされていないので、割り込みハンドラ $\alpha$ が起動される。この時、割り込みB・割り込みC・割り込みDが追加マスクされるので、すべての割り込みがマスクされた状態で割り込みハンドラ $\alpha$ が起動される。

時刻dに割り込みDが発生すると、すべての割り込みがマスクされているので、割り込みハンドラ $\alpha$ は保留状態になり、呼び割り込みハンドラ $\alpha$

の処理に戻る。割り込みハンドラが終了すると、割り込みハンドラ終了手段15が動作し、割り込みマスク状態を割り込みハンドラが起動される前の状態に戻す。この時、割り込みDのマスクが解除され、割り込みハンドラBが保留されているので、割り込みハンドラBが起動される。割り込みベクタによれば、割り込みA・割り込みC・割り込みDが追加マスクされるので、割り込みハンドラBはすべての割り込みがマスクされた状態で動作する。

割り込みハンドラBが終了すると、再び割り込みハンドラ終了手段15が動作し、割り込みマスクを割り込みハンドラB起動前の状態に戻す。この時点では、割り込みハンドラBが保留されているが、割り込みBがマスクされているために、割り込みハンドラBは起動されずに、元の処理に戻る。

時刻eで割り込みマスク変更手段16を使って、割り込みAと割り込みBのマスクを解除すると、割り込みハンドラBが保留されているので、割り込みハンドラBが起動される。割り込みハンドラ

Bは資源1を使うが、通常ルーチンの資源1の操作は終わっているので問題はない。

これらの動作は、資源2、資源3、資源4に対して併他制御を行う時も同様である。つまり、それぞれの資源を使用する割り込みのマスクをかけても、割り込みハンドラB・C・割り込みハンドラA・B・割り込みハンドラB・Cは動作可能である。

以上のように、この方式では各割り込み処理に対して別々にマスクをかけることができる。このため、使用しようとする資源を使う割り込み処理のみをマスクする事が可能となる。その資源に関係のない割り込み処理はマスクされず、従って不必要に延滞する事はない。

#### 【免責の効果】

以上説明したように、本発明の割り込み制御方式を使えば、割り込み処理の延滞による応答性能などの犠牲を必要最小限におさえた、通常ルーチンと割り込み処理ルーチンとの間の資源併他制御が可能となる。

#### 4. 図面の簡単な説明

第1図は本発明の一実施例による割り込み制御方式が適用される計算機システムの構成を示すブロック図、第2図は第1図中の割り込みマスク判定手段の動作を説明するためのフローチャート、第3図は第1図中の割り込みハンドラ起動手段の動作を説明するためのフローチャート、第4図は第1図中の割り込みハンドラ終了手段の動作を説明するためのフローチャート、第5図は第1図中の割り込みマスク変更手段の動作を説明するためのフローチャート、第6図は本発明の割り込み制御方式の動作を説明するために使用される割り込みベクタの一例を示す図、第7図は本発明の割り込み制御方式の動作を説明するために使用される各割り込みハンドラが使用する資源の一例を示す図、第8図は本発明の割り込み制御方式の動作を説明するための図である。

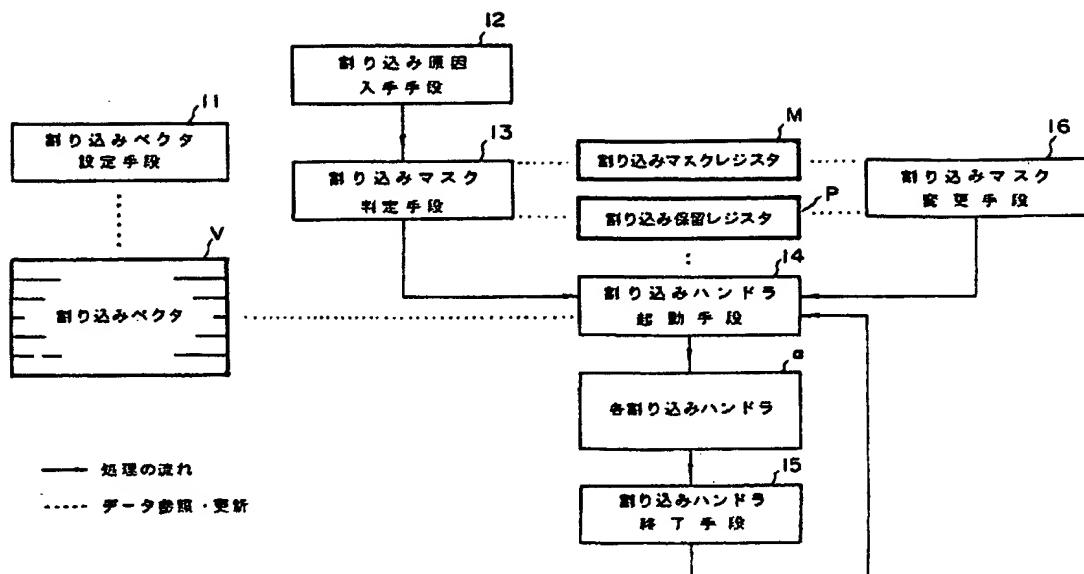
11…割り込みベクタ設定手段、12…割り込み原因入手手段、13…割り込みマスク判定手段、14…割り込みハンドラ起動手段、15…割り込

みハンドラ終了手段、16…割り込みマスク変更手段、V…割り込みベクタ、M…割り込みマスクレジスタ、P…割り込み保留レジスタ。

代理人 (7783)弁理士 池田憲保

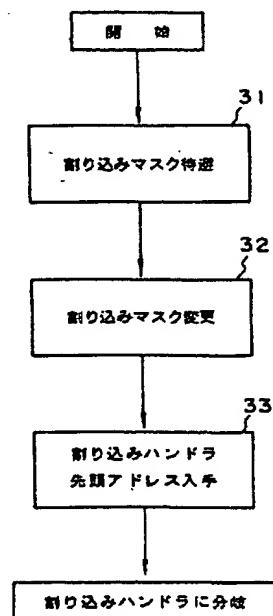
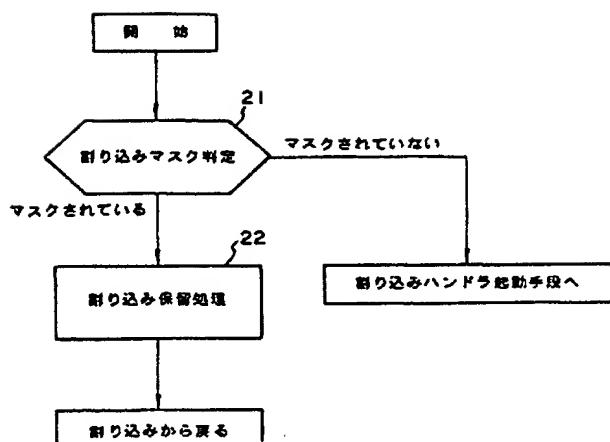


第 1 図

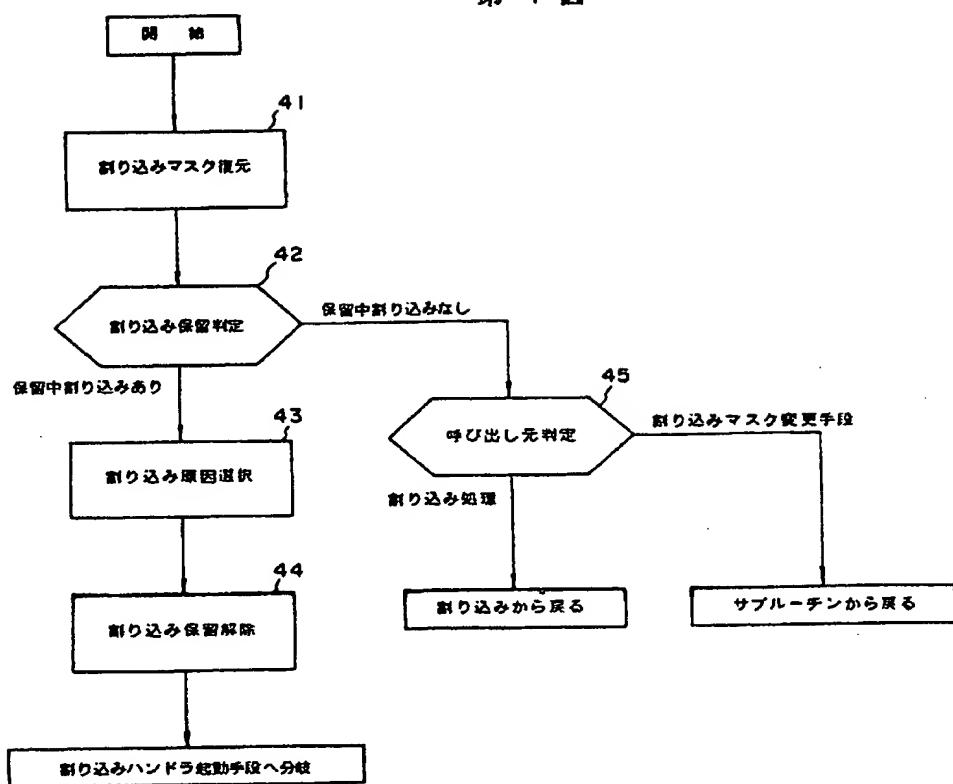


第 3 図

第 2 図



第4図



第5図

